

# Start Document – Personal Finance Tracker

---

## Project Description

The Personal Finance Tracker is a desktop application built with C# and WPF using the .NET 6/7/8 framework. It is designed to help users manage their personal finances by tracking income, expenses, and budgets over time. The application supports transaction categorization, summary visualizations and persistent local storage.

The main goals are to:

- Provide a user-friendly UI for managing financial data
  - Apply best practices in software architecture (MVVM)
  - Demonstrate core concepts such as **data binding**, **commands**, and **data persistence**
- 

## Project Schedule

A schedule will be made to gatekeep the progress of this project. This schedule contains a set of activities from a broad perspective, to visualize what we will be working on, every week until the deadline. A schedule isn't always perfect from the start, so there might be some errors and small corrections along the way, to meet our goal at the end of the road.

This project started on Wednesday, the 4<sup>th</sup> of June 2025, and will end on Friday, the 27<sup>th</sup> of June 2025.

Dates* →	4 <sup>th</sup>	5 <sup>th</sup>	6 <sup>th</sup>	9 <sup>th</sup>	10 <sup>t</sup> h	11 <sup>t</sup> h	12 <sup>t</sup> h	13 <sup>t</sup> h	16 <sup>t</sup> h	17 <sup>t</sup> h	18 <sup>t</sup> h	19 <sup>t</sup> h	20 <sup>t</sup> h
Practice in C#													
Start document													
Development													
Testing													
Presentation (video)													

\* Dates in the month of June

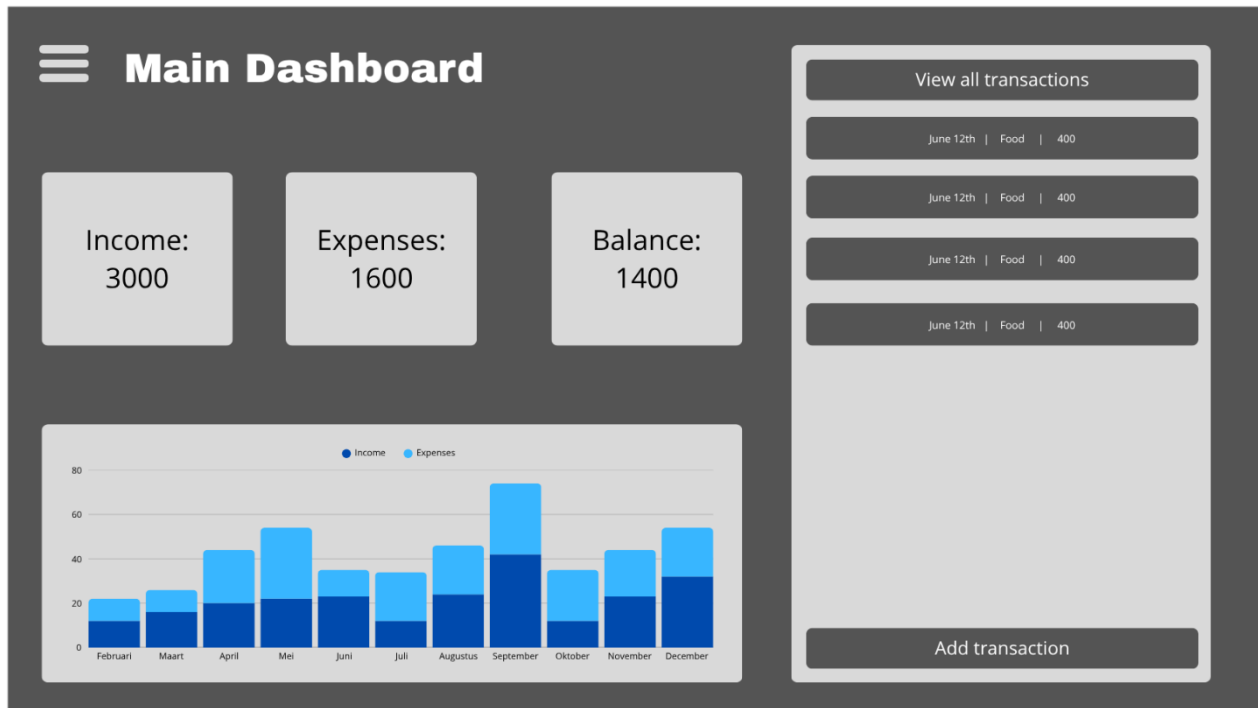
---

## Requirements Analysis (MoSCoW Method)

Priority	Requirement
<b>Must</b>	Users can add, edit and delete income/expense transactions
<b>Must</b>	Data is stored locally using JSON
<b>Must</b>	Use of MVVM pattern for clean separation of concerns
<b>Must</b>	Application uses WPF with data binding
<b>Should</b>	Categories are configurable and color-coded
<b>Should</b>	Budget limits can be set per category
<b>Should</b>	Summary screen shows balance and totals for the month
<b>Should</b>	Basic chart for expense distribution (e.g., pie chart)
<b>Could</b>	Export to CSV or PDF
<b>Could</b>	Support for multiple user profiles locally
<b>Could</b>	Dark/light mode toggle
<b>Won't</b>	Cloud sync or multi-device sync
<b>Won't</b>	Real-time currency conversion or online banking integration

---

## Screens



Main dashboard with an overview of all the expenses and income.

The "Add or edit transaction" screen has a dark gray background. On the left is a light gray form with three sections: "Enter the amount of the expense:" with a text input field, "Category:" with a dropdown menu, and "Description:" with a large text area. On the right is a date picker widget titled "Date". The date picker shows the year "2016" and the date "Fri, Apr 1". Below the date is a calendar grid for April 2016, with the 1st highlighted in a green circle. At the bottom of the date picker are "CANCEL" and "OK" buttons.

Screen that allows the user to add or edit transactions.

Transactions			
Sort by:	Date	Category	Amount
24th of June		Food	400   Burger King
24th of June		Food	400   Burger King
24th of June		Food	400   Burger King
24th of June		Food	400   Burger King

Transaction list where the user can see all her or his transactions.

Budget overview		
Category	Budget limit	Budget progress
Food !	500	<div></div>
Home	800	<div></div>
Free time	300	<div></div>

Budget screen that shows much has been spent on every category.

## Settings

Export data

Import data

Toggle theme:

Currency: Euro

Reset

Export as:

JSON

Note: this deletes all saved data, categories all have to be added by hand again

Settings page where the user can adjust a couple of basic settings and export/import data.

## Class Diagram

